UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Faculty of Engineering,
Built Environment and
Information Technology

Fakulteit Ingenieurswese, Bou-omgewing en
Inligtingtegnologie / Lefapha la Boetšenere,
Tikologo ya Kago le Theknolotši ya Tshedimošo

Study Guide

Department of Chemical Engineering

Process Control

CPB410

2020

Contents

1 Introduction

1.1 Educational approach

The general objective with this module is to emphasise **understanding** rather than memorising, in order to stimulate **creative thinking** and the development of **innovative skills** amongst students in the field of chemical process control. A problem-driven approach to learning is followed. Student-centred and co-operative learning and teaching methods are applied during lectures, tutorial classes in order to optimally develop the above mentioned skills.

You are expected to participate in discussions during lectures and tutor classes. As your fellow students are dependent on the inputs you make, your participation is crucial. After all, you are also dependent on their contributions.

	Name	Office	☎	✉
Lecturer	Carl Sandrock	Eng I, 8-19	012-420-2197	carl.sandrock@up.ac.za
TA	Justin Phillips	ENG II Room 1-56	-	u15063659@tuks.co.za

1.1.1 Consulting hours

You may consult the lecturer and the assistants by appointment only. This policy also holds before tests and exams.

1.1.2 Source material

The following textbook is prescribed. This book will be used for the duration of the module and all students should have a copy. It is recommended that you bring your book to lectures.

> *Seborg, DE, Edgar, TF, Mellichamp DA, Doyle, FJ III, Process Dynamics and Control, Wiley, 3rd Edition, 2011.*

Access to the following textbooks is recommended, but they are not prescribed. These books contain the content of the module and will be useful to broaden your insight. Each study module will have references to the prescribed book as well as to the recommended book.

> *Luyben, W.L., Process Modelling, Simulation and Control for Chemical Engineers, second edition, McGraw-Hil,l 1990.*
>
> *Seborg, D.E., Edgar, T.F., Mellichamp, D.E., Process Dynamics and Control, Wiley, 2^{nd} Edition, 2004.*

1.1.3 Contact time

The CPB 410 module has a weight of 16 credits, which indicates that you should spend approximately 160 hours on the module (preparation for tests, assignments and exams included). The contact time per week is approximately 7 hours (4 lectures per week and 3 tutorial periods) over 12 lectured weeks. This leaves about 76 hours for self-study (about 6 hours per week).

The lecture schedule contains sections in the textbook which should be read before attending the associated lecture. The reading should take about 20 minutes each day. Lectures are presented mostly as a clarification of the preceding reading.

You are advised to take notes by hand during lectures as this has been shown to increase retention. Most of the material is adequately referenced in the study manual and is available in the prescribed text book. For specific topics, additional notes will be provided.

1.1.4 Practical sessions

The practical sessions will take place in a single venue which will be provided with ample power points for plugging in laptops. All students should have access to a laptop for these sessions.

Relevant problems will be handed out before the tutorial sessions. You will be expected to complete most of the problems before the tutorial session and use the time during the tutorial to discuss specific problems. Only final answers will be provided for the tutorial problems and no worked out solutions will be given. Take note that from time to time you will be expected to hand in tutorials for marks. This will be unannounced. The mark will be part of the class test mark that forms part of the final semester mark.

During the practical sessions you will use **Python** as programming language, but may also encounter other modelling environments and languages (for example Modelica and Unisim), to study relevant problems. You must be comfortable with the Python work environment. Computer problems will have to be handed in for marks and will contribute to the class test mark. Also take note of the arrangements regarding the computer project later.

1.1.5 Project

The computer project is a key component of the module and contributes 30% to the final semester mark. The description and expectations of the project are given in the study component of this guide. Please note that the project will be discussed and developed during lectures and tutorials.

1.2 Rules of assessment

Also see the examination regulations in the yearbook of the Faculty of Engineering, Built Environment and Information Technology.

In order to pass this module you must

1. obtain a final mark of at least 50% and
2. obtain at least 40% for the project.

The final mark is calculated as follows

Semester mark	50%
Examination mark	50%

Where the semester mark is calculated as follows

Class tests and tuts	10%
Semester test 1	30%
Semester test 2	30%
Project	30%

This is shown in Figure 1

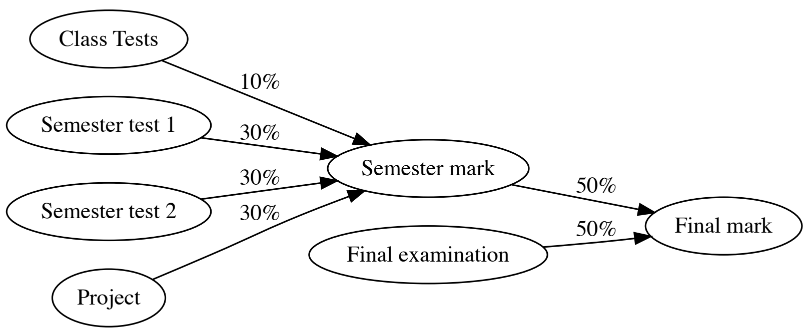

Figure 1: Calculation of final mark

1.2.1 Class tests

Unannounced class tests of approximately 25 min will be written during some of the lecture or tutorial periods, the marks of which will contribute to the class-test mark. Any absence from a class test without a valid reason, will result in zero marks.

1.2.2 Semester tests

Two tests of 90 minutes each will be written during the scheduled test weeks of the School of Engineering. Dates, time and venues will be announced as soon as the timetables become available. The semester tests will take place in a venue where a computer is available.

1.2.3 Absences and sickness

Absences must be supported by an official and valid statement (e.g. a medical certificate) and must be submitted to the lecturer via ClickUP within three days after the date of the test.

There will be one **sick test** of 1 hour and 30 minutes covering the same material as the two semester tests combined scheduled in the week after the second semester test.

1.2.4 Examination

The final examination will be 3 hours long and will be written in a venue where a computer is available.

Unlike the semester tests, absences from an exam due to sickness must be taken up with the administrative staff as set out in the yearbook.

1.3 Plagiarism

Plagiarism poses a considerable problem for academic institutions worldwide as many students and/or researchers do not realise that it is a serious form of academic misconduct which can lead to expulsion from an institution such as a university, civil claims and even criminal charges.

Note the following:

1. Irrespective of whether plagiarism occurred with or without the intent to plagiarise, it remains plagiarism.
2. Plagiarism is a form of misconduct for which a student may be disciplined in terms of the University's disciplinary code (when an assignment is submitted for assessment, plagiarism is the equivalent of cheating in an exam).
3. Plagiarism is also not dependent on the verbatim presentation of a specific number of words, as, under certain circumstances, the use of a single phrase or idea can be judged to constitute plagiarism.

Copies of the document on plagiarism are available from ClickUP and the Library Website mainpage: http://www.library.up.ac.za/plagiarism/index.htm

1.4 Grievance procedures

All grievances must be submitted in writing with specifics of the incident or the nature of the complaint. It is imperative that you follow the procedure outlined below in order to resolve your issues:

1. **Consult the lecturer concerned** about your grievances/concerns.
   If the matter has not been resolved,

2. **consult the class representative** (The primary function of the Class Representative is to serve as a two-way communication channel between the class and the lecturer).

   If the matter has not been resolved,
3. **consult the module co-ordinator** (large modules with multiple lecturers)

   If the matter has not been resolved,
4. **consult the Head of Department**.

   If the matter has still not been resolved,
5. **consult with the Dean of the Faculty**

2 Study Component

2.1 Module objectives, articulation and learning outcomes

2.1.1 General objectives

In this module you must develop the necessary skills to design controllers and understand the main principles of process control. Feedback control, feedforward control, advanced control, model predictive control and discrete control are addressed. The tools developed in **CPN 321 Process Dynamics 321** are extended to allow for stability and performance analysis of closed loop systems in the time, Laplace, frequency and z-domains. After completing the module the student will be able to:

1. Understand the principle of feedback control in the Laplace, Frequency and time domains and be able to design feedback controllers using relevant techniques and to analyse the behaviour of such systems.
2. Understand and apply the major controller tuning techniques.
3. Be able to define performance measures and analyse the performance of a feedback controller
4. Understand the principle of feedforward control and design and analyse feedforward controllers
5. Understand the principle of model predictive control and the internal model control structure.
6. Use discrete (z-domain) analysis to design and understand discrete controllers.
7. Apply the basic process identification techniques to obtain a process model.
8. Understand the challenges associated with multivariable control and apply basic techniques to alleviate them
9. Understand the importance of addressing control on a plant-wide basis
10. Develop a large-scale control strategy
11. Document control strategy using a P&ID

2.1.2 Prerequisites for CPB410

You must have complied with the CPN 321 GS requirement, meaning an average of 40 % for the semester mark and exam mark combined.

2.1.3 Critical learning outcomes

ECSA (Engineering Council of SA) is an external body that ensures that graduate engineers receive training the nature of which will enable them in time to conform to the requirements for registration as Professional Engineer. To achieve this, ECSA ratifies that the undergraduate curriculum conforms to relevant and realistic requirements by, amongst other things, regular visits to tertiary institutions offering prescribed programmes. One of these requirements is that the programme must comply with 11 critical learning outcomes.

The following ECSA exit-level outcomes are addressed in this module, i.e. at the conclusion of this module you will be capable of:

ECSA 2.1: Problem solving Identify, formulate, analyse and solve complex engineering problems creatively and innovatively.

ECSA 2.2: Application of scientific and engineering knowledge Typical problems will require the application of basic chemical engineering principles like mass and energy balances as well as the extensive use of calculus.

ECSA 2.3: Engineering design Students are required to develop and design a control system for a real system using a variety of control system design techniques. This design incorporates most of the control theory in CPB 410.

ECSA 2.5: Engineering methods, skills, tools, including information technology

This module (and especially the project) requires basic programming skills (**Python**) to model more complex systems.

2.2 Study themes

2.2.1 Conventional feedback control

On completion of this study theme you should be able to do the following:
From Seborg, Chapter 1

1. Understand the need for process control and completely understand the basic nomenclature of control systems, specifically the controlled variable, the manipulated variable and the disturbance or load variable for a specific system.
2. Understand the hierarchy of process control activities.

3. Understand that feedback control is a corrective and not an anticipatory action

From Seborg, Chapter 7

4. Understand what is meant with proportional control and that the magnitude of the proportional gain will influence the speed of response of the controller.
5. Determine the sign of the proportional gain of the feedback controller and how this affects control action. (ie direct and reverse, or indirect action)
6. Understand why proportional control alone can result in an offset.
7. Be intimately familiar with the three basic types of control (proportional, integral and differential) and understand why/how each type assists in the controller "decision making".
8. Understand and be able to show that integral control action will continue until the error is eliminated (or until the final control element reaches a saturation point)
9. Explain why differential control will accelerate the control action.
10. Express the P, PI and PID algorithms in the time and Laplace domains
11. Understand the disadvantages of on-off controllers
12. Obtain responses (qualitatively and quantitatively) for setpoint changes and load changes.

Study material: Seborg Chapter 1, §7.1-7.5

Number of lecture periods: 6

2.2.2 Laplace domain analysis of control systems

On completion of this study theme you should be able to do the following:

1. Represent the feedback control loop in block diagram format and realise that the variables are represented as deviation variables.
2. Derive the transfer functions relating a setpoint change to the controlled variable (servo problem), and a change in the disturbance variable to the controlled variable (regulatory problem).
3. Show that the denominator of the servo and regulatory transfer functions of a feedback loop are the same (characteristic equation).
4. Clearly understand that the roots of the characteristic equation contain information on the stability and shape of the servo and regulatory responses.
5. Understand that the controller parameters (K_c, τ_I, τ_D) determine what the roots of the characteristic equation will be.
6. Realise that different controllers (type and parameters) will have different roots of the characteristic equation (poles of the closed loop transfer function).
7. Understand and apply the Routh stability criterion.

Study material: Seborg Chapter 10

Number of lecture periods: 4

2.2.3 PID controller design, tuning and troubleshooting

On completion of this study theme you should be able to do the following:

1. Use model-based controller design methods
   (a) Direct synthesis method
   (b) Internal Model Control (IMC)
2. Approximate the above controllers as PID controllers (find controller parameters)
3. Know common performance criteria for closed-loop systems
4. Determine PID controller parameters using the following techniques:
   (a) IMC based relations
   (b) Time integral performance measures
   (c) Cohen and Coon technique (note that the open loop reaction curve does not contain the controller transfer function).
   (d) Overshoot, decay ratio, settling time
   (e) Ziegler Nichols technique
5. Have a basic understanding of applications for common control loops

Study material: Seborg Chapter 11

Number of lecture periods: 6

Semester test 1 will cover study themes 1, 2 and 3.

2.2.4 Frequency domain analysis of control systems

Before embarking on this study theme you should verify that you are able to do the following:

- Derive and apply the substitution rule ($s = i\omega$).
- Be able to represent frequency responses as Nyquist or Bode plots and interpret them correctly.
- Appreciate the benefit of the log scale in handling systems in series.

Study material: Seborg, §13.1-13.3, §13.5

On completion of this study theme you should be able to do the following:

1. Apply the Bode- and Nyquist stability criteria
2. Know the definitions of the gain and phase margin
3. ~~Design controllers by allowing for phase and gain margins~~
4. ~~Design controllers by using the maximum log modulus of the closedloop frequency response~~

Study material: Seborg §13.4-13.7

Number of lecture periods: 3

2.2.5 Advanced control methods

On completion of this study theme you should be able to do the following:

From Seborg, Chapter 14:

1. Understand the difference between feedforward and feedback control.
2. Design feedforward controllers and understand the physical realisability of such controllers.
3. Understand and apply ratio control.
4. Understand the advantages of the combination of feedforward and feedback control and apply this method.

From Seborg, Chapter 15:

5. Understand and apply the advantages of cascade control.
6. Understand the effect of dead-time on a feedback loop and apply methods to compensate for this – the Smith predictor.
7. Have a basic knowledge of the concepts inferential control, selective control and split-range control.

Study material: Seborg, Chapter 14, §15.1-15.4

Number of lecture periods: 8

2.2.6 Discrete control and analysis

Before embarking on this study theme you should verify that you are able to do the following:

- Take note of the sampling theorem and the factors affecting the choice of sampling interval.
- Mathematically represent the discretisation of continuous signals.
- Understand the working of hold-elements and represent this mathematically.
- Discretise continuous models.
- Know and apply the definition, properties and inversion of the z-transform.
- Calculate the discrete response of dynamic systems.
- Derive pulse transfer functions. Understand the physical realisability of pulse transfer functions.
- Give the qualitative response of discrete systems.

Study material: Seborg, §17.1-17.3

On completion of this study theme you should be able to do the following:

1. Know the elements of digital control loops.
2. Derive the closed loop transfer functions of control loops containing discrete elements.
3. Perform stability analysis of discrete systems and understand the role of the sampling interval.

4. Understand and apply the discretisation of PID control algorithms (velocity- and position form).
5. Understand the origin of ringing poles and take the necessary steps to move or remove them.
6. Understand and apply
   (a) Dahlin's control algorithm.
   (b) The Vogel-Edgar Algorithm
   (c) Digital IMC

Study material: Seborg, §7.6 (Discrete PID), §17.3-17.5

Number of lecture periods: 7

2.2.7 Multivariable control

Before embarking on this study theme you should verify that you are able to do the following:

- Basic operations on matrices. Review your linear algebra course notes
- Represent multivariable systems using the following representations (Seborg Section 5.5 and 5.6)
  - Transfer function matrices
  - State space representation

On completion of this study theme you should be able to do the following:

1. Understand the concept of interaction in multivariable systems
2. Identify controlled, manipulated and disturbance variables in multivariable systems
3. Understand the problem of pairing controlled and manipulated variables in multivariable systems
4. Calculate and apply the RGA to interaction analysis
5. Find good pairings for decentralised control
6. Determine decouplers for multivariable systems
7. Understand the MPC algorithm (you do not have to be able to implement an MPC controller).
   (a) Understand that MPC uses the model predictions and optimisation at every time stemp
   (b) Understand the basic terminology of receding horizon control (prediction horizon, control horizon, objective function, bias).

Study material: Seborg, §16.1-16.2, 16.5-16.6, §20.1-20.2

Note Seborg Chapter 20 is available online from http://www.wiley.com/go/global/seborg

Number of lecture periods: 8

Semester test 2 will cover study themes 4, 5, 6 and 7. Note that this will assume understanding of concepts from previous themes.

2.2.8 Control practice

Before embarking on this study theme you should verify that you are able to do the following: From Seborg, Chapter 8:

- Have basic understanding of the sensors, transmitters and transducers available for measuring common variables of interest to Chemical Engineers
- Understand the operation of pneumatic control valves
- Be able to size a control valve and select the operating characteristic to linearise the installed characteristic.
- Understand the terms *resolution* and *precision* used to describe instrument accuracy.
- Know the standards associated with signal transmission

On completion of this study theme you should be able to do the following:

From Seborg, Chapter 8:

1. Select the operating mode (A/O vs A/C) of a control valve based on safety considerations
2. Be aware of dynamic measurement errors

From Seborg, Chapter 9:

3. Be aware of the importance of safety as it pertains to process control activities
4. Be aware of alarms and interlock systems
5. Understand that alarms need to be included rationally, not as a default option

From Seborg, Chapter 12

3. Understand that process units will have more than one control loop
4. Be able to calculate the control degrees of freedom for a unit
5. Choose good controlled and manipulated variables for process units

From Seborg, Appendix F and G:

6. Understand the unique issues which arise when process units are combined
7. Be able to avoid overspecification on control systems featuring multiple units
8. Be able to analyse large-scale control systems qualitatively

From Seborg, Appendix I and P&ID handout:

9. Know the ISA standard symbology for P&IDs
10. Know the broad standards of what belongs in PFDs, P&IDs and MFDs
11. Be able to read a P&ID
12. Be able to draw a P&ID

Flow diagrams Instrumentation symbols; Development of a P&ID

Study material: Seborg Chapter 8, §9.1-9.2, §12, Appendices F, G, I and P&ID handout

Number of lecture periods: 6

The exam will cover all of the study themes.

2.3 Project

The aim of the project is to design and implement controllers for a **non-linear** process. The project is split into two group projects and one individual project. You will be divided into groups and each group will receive a temperature control lab unit. *The project will be executed in **Python** as the programming language.*

The development of the relevant parts of the project will be done during the computer tutorials. Part 3 will be a time-limited individual assessment. Each student will be expected to develop and run a numerical simulation of this system and submit results under exam conditions.

2.3.1 Part 1

What you will get You will receive a working temperature control lab system. The system is based on the Arduino platform and contains two heating units with two temperature sensors. The tclab Python package can be used to interact with the lab unit.

You will also receive previous projects and be called upon to select one project to critique.

Coding expected of you Simulate a physically realisable continuous PID-controller and have it write to the physical system. Also develop a FOPDT model and simulate the continuous PID acting on the model of the system. Take note of the way that differential control action can be implemented in a practical controller. (Seborg pp. 135).

Your code should make it easy to choose between P, PI, PID and PD modes of operation. Note that each of these modes will have a differently-shaped state space representation.

It is imperative at this stage to integrate your continuous PID controller with a smaller time step than you use to sample the temperature control lab, especially when there is differential action.

Investigation

- With the system at steady-state, apply a step disturbance of an adequate magnitude to the heater input and obtain the system's process reaction curve at a specific operating condition. Remember: The important output is the measured temperature

- Using this response curve, obtain a first order plus dead time approximation of the model. Basically, find parameters K_p, (process gain), τ_p (time constant) and τ_d (dead-time).
- Compare your model to the previous group's model. How closely does their model predict the outputs from your TCLab?
- Quantify the extent of non-linearity of your system by applying step disturbances of various magnitudes in a positive and negative direction relative to the chosen steady-state operating condition. How well did the previous group do in modelling this aspect? Mention the good things and bad things they did in the representation of nonlinearity.
- Investigate the effect of the point of linearisation by repeating the experiment above at a significantly different steady state point. Do you obtain the same gain?
- Using the process parameters obtained from the reaction curve, calculate the Cohen-Coon controller parameters for a P, PI and PID-controller. Applying each, determine the corresponding response curves for a step in the setpoint and a step in the disturbance.
- Making use of any other controller tuning technique (trial-and-error, minimising error integrals, 1/4-decay ratio, etc.) obtain a set of controller parameters which give a better reponse than that obtained using the Cohen-Coon parameters. Make sure you choose a different method than the previous group. Compare their best settings against yours.
- Compare the results obtained for each of the controller types using the different tuning techniques. Plot the results for each type of controller on the same graph.
- What effect does the non-linear nature of your system have on the control of the system? Specifically remember that manipulated variable limits are non-linear effects. Do your conclusions about nonlinearity agree with the previous group's?
- Determine and discuss the effect that **system noise** has on the controllability of the process.

  It will be useful for you to introduce noise into your simulation of the offline system to investigate this.
- You should now be able to comment of the **performance** and **robustness** characteristics of you system. (NB You will have to obtain information regarding definition and quantification of these terms before you can answer this)

  Make sure you make it clear what new research you have done and what you have used from the previous group.
- Discuss the effect of the sampling rate of the TClab system on controller performance. Make sure you vary the rate at which you sample the TCLab for this step.

The minimum requirement for Part 1 That the PID controller controls your controlled variable for load and setpoint disturbances.

Hand-in A report discussing the points above as well as the code which you used to implement the controller.

2.3.2 Part 2

Part 2 tests your ability to implement a controller type other than PID and your ability to implement digital controllers.

Coding expected of you

- Add a discrete approximation of a PID controller to your system and compare its performance with the continuous PID controller of Part 1. Note that the controller algorithm can be written in the so-called **velocity form** or in the **position form**. Implement both forms.
- Add a continuous Internal Model Controller (IMC) based on a FOPDT model to your system. Do not approximate the IMC controller as a PID controller – implement the model and controller block separately.
  Tune the controller (by choosing τ_C) and compare its performance with that of the PID-controller of part I of this project. Do you agree with the tuning the previous group obtained?

Discussion When discussing the results obtained based on the above implementation, be sure to mention:

- The behaviour of the feedback PID controllers (discuss different choices for parameters and P, PI and PID modes).
- Effect of non-linearities on controller performance.
- Effect of noise on controller behaviour.
- Effect of sampling rate on the behaviour of the discrete controllers
- The differences between velocity and position form
- The differences between continuous and discrete controllers
- Limitations/advantages of model-based controllers.
- Propose norms for the evaluation of controller performance on this system. Compare the controllers you have implemented based on these measures.

For each of the points above, mention the results and conclusions the previous group reached and evaluate them critically by comparing with your own conclusions.

Hand-in A report discussing the points above as well as the code which you used to implement the controllers.

2.3.3 Part 3

Part 3 is an individual assignment assessed during a three hour practical session under **closed book** examination conditions in a computer laboratory.

You will be given :

- an **unseen simulation** of a system in the form of a Jupyter notebook
- the standard Jupyter notebook cheat sheet
- The standard data sheet
- A computer with the Anaconda Python distribution installed as well as the tbcontrol and python-control libraries.

You will need to

- understand the code well enough to answer questions about the implementation
- implement one of the controllers from the previous parts during the session.
- use the insights you gained from the other parts to make sensible decisions about sampling rates, model parameters and/or controller tuning.

You will not receive the simulation code before the session. You will not be allowed access to the Internet, previous submissions by other groups or your own.

This part is designed to ensure that you do not delegate the coding part of the project to one member of your team and avoid learning the implementation details.

2.3.4 Assessment

The project mark will be the average of the marks obtained for the three parts (33.3 % each).